

Dclick Dojo 1

Conceitos, não notas

Problema:

A escola Dclick está mudando de cara e seguindo as novas tendências no ensino para melhor atender seus alunos. Por esse motivo, não serão mais atribuídas notas aos alunos, e sim conceitos. O problema é que os professores ainda não conseguem fazer facilmente tal mudança, por isso pediram a você (programador) que criasse um programa que faça tal atribuição automaticamente.

Os professores foram gentís, e criaram uma interface Java para ajudar na solução, contendo os métodos que serão utilizados:

```
public interface GerenciadorConceitos {  
  
    boolean adicionaConceito(Operador operador1, double valor1,  
        Operador operador2, Double valor2);  
  
    String atribuiConceito(double valor);  
  
}
```

Para o primeiro método:

Operador é um enum com os possíveis valores: *MENOR*, *MENOR_IGUAL*, *MAIOR*, *MAIOR_IGUAL*

Operador descreve a comparação com o valor que é passado em seguida como parâmetro, por exemplo, MENOR 3.0 é válido quando o valor que está sendo comparado é menor que 3.

O segundo valor de operador pode ser nulo. Dessa forma estaremos cadastrando um intervalo com apenas uma restrição. Caso o valor em 'operador2' seja nulo, o valor em 'valor2' deve ser descartado.

Os valores em 'double' podem variar entre 0 e 10, com incrementos de 0.1, ou seja, o intervalo com todos os números é 0.0, 0.1, ... , 9.9, 10.0.

O método será invocado 3 vezes consecutivas, sendo que a primeira servirá para cadastrar o conceito RUIM, a segunda para o conceito REGULAR e a terceira para o conceito BOM, nesta ordem respectivamente.

Caso um intervalo seja inválido, o método deve restorna 'false', caso contrário deve ser retornado 'true'.

Um intervalo é considerado inválido, se este sobrepõe um outro intervalo já cadastrado. Mesmo após ocorrer erro, o cadastro deve proseguir normalmente.

Sempre será cadastrado o intervalo inteiro, de 0.0 a 10.0 e nunca será cadastrado um intervalo que se sobreponha, por exemplo $x \leq 3$; $x \leq 5$.

Para o segundo método:

Após o cadastro bem sucedido dos 3 conceitos, será chamado para atribuir conceitos aos valores que forem passados como parâmetro, baseado no cadastro feito anteriormente.

Objetivo:

Implemente baseado na interface, um programa que resolva o problema dos intervalos de notas para todos os casos.

Exemplo de uso:

```
adicionaConceito(MENOR, 3.0, null, null) -> true
adicionaConceito(MAIOR_IGUAL, 3.0, MENOR, 7.0) -> true
adicionaConceito(MAIOR_IGUAL, 3.0, MENOR, 7.0) -> false
adicionaConceito(MAIOR_IGUAL, 7.0, null, null) -> true

atribuiConceito(0.0) -> "RUIM"
atribuiConceito(3.0) -> "REGULAR"
atribuiConceito(7.0) -> "BOM"
```